

RTS게임을 위한 지능 아키텍처 설계 및 평가: 스타크래프트 사례연구

조호철⁰, 김정중
세종대학교 컴퓨터공학과
(chc2212@naver.com, kimkj@sejong.ac.kr)

Design and Evaluation of Intelligence Architecture for RTS Games: Case Study of Starcraft

Ho-Chul Cho⁰, Kyung-Joong Kim
Dept. of Computer Engineering, Sejong Univ.

요약

RTS게임은 실시간으로 진행되고, 여러 가지 요소들이 복잡하게 연결되어 있기 때문에 지능형 시스템을 만들기 쉽지 않다. 본 논문에서는 RTS게임의 지능 아키텍처 구조도를 제안하였고, 이것을 실제 스타크래프트 게임에서 구현하였다. 지능형 아키텍처를 기반으로 한 인공지능 봇 Xelnaga는 전문가 시스템 (Rule-Based) 기반으로 만들었으며 스타크래프트 인공지능 봇이 서로 대결하는 대회인 IEEE CIG, AIIDE Starcraft AI Competition에서 우수한 성적을 거두었다.

1. 서론

RTS(Real-Time Strategy Game)는 1994년 워크래프트를 시작으로 대중화되었고, 1998년 스타크래프트로 인해 주요 게임 장르 중 하나로 자리 잡게 된다. 그러나 짧은 역사와 장르의 특성상 발생하는 복잡성으로 인해 해당 분야의 인공지능에 대한 연구는 미진하였다. 2010년부터 시작한 대회인 IEEE CIG(Computational Intelligence and Games)와 AIIDE(AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment) Starcraft AI Competition은 스타크래프트 인공지능 봇을 만들고, 봇이 서로 대결하는 대회이다. 이 대회를 통해 사람과 비슷한 수준의 RTS게임을 하는 것을 목표로 하는 여러 인공지능 봇들이 나오고 관련연구가 활발히 진행 중이다. 본 논문에서 소개할 Xelnaga도 대회에 참가하기 위해 개발을 하였고, 대회에서 우수한 성적을 거두었다.

2. Xelnaga

2.1. Xelnaga 소개

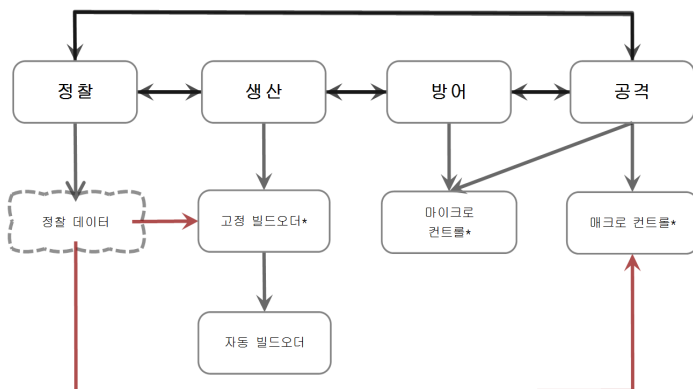


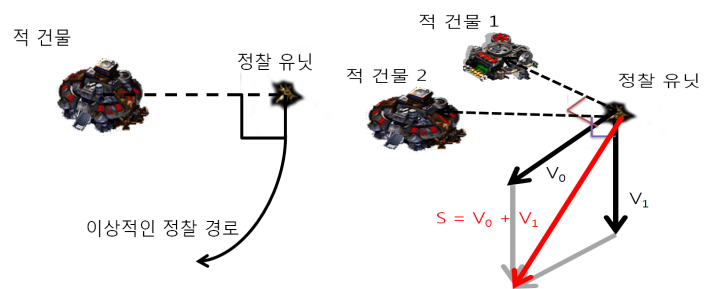
그림1: RTS게임의 기능에 따른 지능형 아키텍처 구조도

* 빌드오더 : 건물을 짓거나 유닛을 뽑는 순서
* 마이크로 컨트롤 : 소규모 유닛들의 컨트롤
* 매크로 컨트롤 : 대규모 유닛들의 컨트롤

스타크래프트와 같은 RTS게임의 기능들은 그림 1과 같이 분류되며 이것을 바탕으로 지능 아키텍처를 설계할 수 있다. 정찰을 하여 적의 정보를 알아내고, 이를 통한 생산, 방어, 공격이 서로 상호작용하며 이루어지는 것이 일반적인 RTS게임의 구조이다. 위와 같은 4가지 기본적인 요소들을 바탕으로 아키텍처를 구성할 수 있다. 일단 정찰을 하여 정보를 얻고 이 정보를 통해 빌드 오더를 바꾸거나 공격타이밍을 바꾼다. 그리고 생산의 경우 경기초반의 일정부분은 미리 입력된 빌드오더를 사용하고, 이후에는 자동으로 빌드오더를 만들어낸다. 방어와 공격에서는 크게 마이크로컨트롤과 매크로 컨트롤로 나누어 구현할 수 있다. Xelnaga는 위와 같은 아키텍처로 구성되었고, 전문가 시스템 (Rule-Based)에 기초한 인공지능 알고리즘을 사용한 스타크래프트 봇이다.

2.2. 초반정찰

RTS게임에서 정찰은 필수요소이며 경기 초반에 적 지로 정찰 유닛을 보내서 정보를 얻어야 한다. Xelnaga는 정찰기능을 구현하였으며 적기지에서 최대한 오래 살아남는 것을 목표로 알고리즘을 만들었다.



(a) 적 건물이 하나인 경우 (b) 적 건물이 여러 개인 경우

그림 2: 적 건물 개수에 따른 정찰유닛의 움직임

그림 2(a)와 같이 적의 건물이 하나밖에 없을 경우 이상적인 정찰루트는 원을 돌면서 정찰하는 것이다. 원을 돌면서 정찰함으로써 건물의 시야를 계속 확보하면서 적의 공격을 피할 수 있다. 적의 건물이 만약 여러 개라면 위의 알고리즘을 확장하여 적의 건물 주위를 원을 도는 것처럼 움직일 수 있다. 그림 2(b)와 같이 각 건물들이 정찰유닛과 직각을 이루는 벡터를 $V_0, V_1, V_2, \dots, V_n$ 이라 하면 프로브(정찰병)가 이동하고자 하는 방향 S 는 모든 벡터의 합과 같다.

$$S = \sum_{i=1}^N V_i$$

이러한 초반정찰은 상대방의 정보에 따라 빌드오더를 변화시키는데 사용되었고, 실제 플레이어를 대상으로 실험하여 적의 빌드오더를 예측하는 방법을 제안하였다 [1].

2.3. 생산 및 방어

경기 초반에는 미리 입력해놓은 빌드오더를 사용하고, 중, 후반에는 자동으로 빌드오더를 만들어서 사용한다. 또한 초반정찰을 통해 얻은 정보를 활용하여 고정 빌드오더를 수정하는 시스템을 가지고 있고, 상대방의 공격에 대해 적절히 대응하는 방어시스템을 갖추고 있다.

a. 경기 중 정찰정보를 활용한 빌드오더의 수정

상대가 프로토스일 경우 초반 일꾼정찰을 통해 얻은 정보를 바탕으로 빌드오더가 수정된다. 고정 빌드오더를 통해 사이버네틱코어까지 건설명령을 내리고, 사이버네틱코어가 완성되는 시점에 상대방 건물의 종류와 수에 따라서 세 가지로 빌드오더로 변경된다.

표 1: 상대방 건물의 종류와 수에 따라서 달라지는 세 가지 빌드오더 ($\alpha=3$, $\beta=2$)

매우 빠른 공격	Assimilator = 0, Gateway $\geq \alpha$
빠른 공격	Assimilator = 0, Gateway = β
일반	위의 규칙에 해당되지 않는 상황

세 가지 빌드오더는 표 1에 나타난 규칙에 따라서 변경된다(Assimilator와 Gateway는 스타크래프트의 건물 이름임). 각 경우마다 다른 고정 빌드오더가 사용되고, 자동빌드오더로 넘어가는 시간이 달라진다.

b. 자동 빌드오더

중, 후반에는 많은 변수가 발생하므로 고정된 빌드오더로 생산하는데 한계가 있기 때문에 자동 빌드오더를 사용한다. 자동 빌드오더는 생산할 건물, 유닛들(Pylon, Probe 등)에 대해서 각 건물, 유닛들마다 규칙을 사용하여 생산할지 여부를 결정한다. 그리고 Pylon, 고급건물, Probe, Zealot(Dragoon), GateWay의 우선순위에 따라서 어떤 유닛이나 건물을 우선적으로 생산할지 결정한다.

예를 들어 병력 생산 건물인 GateWay같은 경우 모든 GateWay에서 유닛이 생산되고 있으면 GateWay를 추가로 건설하라는 명령이 내려진다. 그리고 우선순위에서 제일 마지막에 있기 때문에 높은 우선순위의 명령이 없을 때 GateWay가 만들어지게 된다.

c. 방어시스템

Xelnaga는 적의 공격유닛에 아군의 공격유닛을 2(아군)대 1(적군)로 할당하는 방어시스템을 가지고 있다. 이것은 적이 한 곳으로 집중공격을 들어왔을 때에도 적절한 방어시스템일 뿐만 아니라 적이 분산해서 공격해왔을 경우에도 효율적인 방어시스템이다.

2.4. 공격

스타크래프트에서 아군과 적군에 대한 전력을 파악하는 것은 쉽지 않은 일이다. 단순히 상대가 가지고 있는 유닛숫자뿐만 아니라 유닛의 배치나 거리에 따라서 전력을 다르게 평가해야 하기 때문이다. 따라서 유닛이 '머쳐있는 정도'를 파악할 수 있는 시스템이 필요하고, 이것을 위하여 자신감 시스템을 만들었다.

자신감시스템의 자신감은 해당유닛 주변에 적 유닛에 비해 아군유닛이 얼마나 많은지 나타내는 척도이다. F

를 유닛의 공격력, v 를 공격속도라고 하면 각 유닛의 자신감 C 는 아래와 같이 계산된다(해당 유닛 주위의 유닛만 계산하며 enemy는 적군, self는 아군을 의미함).

$$C = F_{enemy} \times v_{enemy} - F_{self} \times v_{self}$$

a. 자신감이 낮은 유닛의 회피

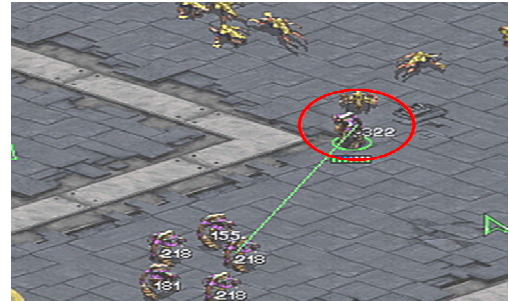


그림 3: 자신감이 낮은 유닛의 회피

자신감이 일정 숫자 이하로 떨어진 유닛은 자신감이 높은 유닛에게 이동한다. 그림 3은 주위에 빨간색 원이 그려져 있는 유닛이 주변의 적 유닛들 때문에 자신감이 낮아져서 자신감이 높은 아군 유닛들에게 이동하는 모습이다.

b. 아군과 적군의 주 병력 판단

자신감을 통해 아군과 적군의 주 병력을 판단할 수 있다. 적의 주 병력과 전투를 하고자 할 때 이것을 활용할 수 있고, 한 번에 여러 적을 공격할 수 있는 하이템플러의 사이오닉스톰과 같은 범위공격 마법을 사용할 때 적의 자신감이 높은 유닛에게 사용하면 효과적으로 공격을 할 수 있다.

3. 성능평가

인간의 심리를 이용하고, 실시간으로 복잡한 상황이 만들어지는 RTS게임의 인공지능 봇을 만드는 일은 쉽지 않은 일이다. 본 논문에서는 지능형 아키텍처를 제안하고, 이것을 활용하여 Xelnaga라는 스타크래프트 인공지능 봇을 만들었다. Xelnaga는 스타크래프트 인공지능 봇끼리 대결하는 Starcraft AI Competition에 참가하여 표 2와 같은 성적을 거두었다.

표 2: Xelnaga의 Starcraft AI Competition 경기 결과

대회	참가팀 수	경기 수	승률	순위
IEEE CIG2011	10	70	51%	3위
IEEE CIG2012	10	810	51%	6위
AIIDE 2012	10	1656	52%	6위

IEEE CIG2011대회는 각각 5팀씩 2조로 나뉘어 예선전을 한 뒤 각 조당 2팀씩 4팀으로 결선을 치르는 방식으로 진행되었다. Xelnaga는 예선에서 1위로 결선에 올라갔고, 결선에서 3위를 하였다. IEEE CIG2012와 AIIDE2012대회는 모든 팀들이 리그형식으로 경기를 하여 승률이 가장 높은 봇이 우승하는 방식으로 진행되었고, 게임진행이 자동화되어 많은 수의 경기가 가능했다. IEEE CIG2012에서 6위(승률 51%), AIIDE2012에서 6위(승률 52%)를 하였다.

참고문헌

[1] H.-S. PARK, H.-C. CHO, K.-Y. LEE and K.-J. KIM, Prediction of Early Stage Opponents Strategy for StarCraft AI using Scouting and Machine Learning, *Workshops at ACM SIGGRAPH ASIA (Gaming Track)*, 2012